Analyzing a Human-based Trust Model for Mobile Ad Hoc Networks

Pedro B. Velloso[1], Rafael P. Laufer[2], Otto Carlos M. B. Duarte[3], and Guy Pujolle[1]

[1]Laboratoire d'Informatique de Paris 6 (LIP6)
UPMC - Paris, France

[2]Computer Science Department
UCLA CA, USA

[3]Grupo de Teleinformática e Automação (GTA)
UFRJ - Rio de Janeiro, RJ, Brazil

Abstract— **This paper analyzes a trust model for mobile ad hoc networks. We provide nodes with a mechanism to build a trust relationship with its neighbors. The proposed model considers the recommendation of trustworthy neighbors and the experience of the node itself. The interactions are limited to direct neighbors in order to scale on mobile networks. The results show the efficiency and the trade-off of our model in the presence of mobility. We also analyze the advantages of considering the relationship maturity, i.e. for how long nodes know each other, to evaluate the trust level. The maturity parameter can decrease the trust level error up to 50%.**

I. Introduction

The main difference between a conventional network and an ad hoc network is the lack of infrastructure. For this reason, nodes accumulate the role of router, server and client, compelling them to cooperate for the correct operation of the network. This peculiar characteristic hinders applications and protocols conceived for conventional networks to perform efficiently in ad hoc networks. Therefore, new specific protocols for this type of network have been proposed and developed. However, the majority of the protocols and applications for ad hoc networks considers the perfect cooperation among all nodes. It is assumed, then, that all nodes behave in accordance with the specifications of the applications and protocols defined for the network. Nevertheless, this assumption may be false, due to resource restrictions or malicious behavior. Consequently, the nodes may not behave as expected by protocols or applications, causing the network to not work properly. Thus, the assumption that nodes behave correctly can lead to unforseen pitfall, such as a low network efficiency, a high resource consumption, and a higher vulnerability to attacks. Therefore, a mechanism that allows nodes to infer the trustworthiness of other nodes is necessary. Providing nodes with a trust level is not only useful when nodes misbehave. In an ad hoc network there is no central entity responsible for configuring, managing, and repairing the stations. According to the paradigm of autonomic networks, a node should be capable of self-learning, self-configuring, and self-managing by means of collecting local information and exchanging information with its neighbors. Thus, it is important to communicate only with trustworthy neighbors, because the exchange of information with compromised nodes can deteriorate the autonomy of ad hoc networks.

Several papers propose trust models for ad hoc networks. Liu *et al.* [1] propose a trust model for ad hoc networks based on the distribution of threat reports. The goal is to make security-aware routing decisions, where nodes use the trust level as an additional metric for routing packets. Nevertheless, they assume that nodes cooperate with each other, which is not always the case. They also assume that all nodes are capable of detecting malicious behavior by means of Intrusion Detection Systems (IDS). This assumption leads to high energy consumption, which is clearly not an appropriate option for ad hoc networks.

Pirzada and McDonald [2] propose another trust model for ad hoc networks to compute the trustworthiness of different routes. Nodes can use this information as an additional metric on routing algorithms. Although the authors present an interesting approach, the model presents several disadvantages. For instance, it is currently restricted to Dynamic Source Routing (DSR) protocol. It also relies on using promiscuous mode ignoring the energy constrains of mobile nodes. Finally, it requires each node to store information for all other nodes in the network, which is clearly non-scalable.

Virendra *et al.* [3] present an trust-based architecture that allows nodes to make decisions on establishing criptographic keys with other nodes and forming groups of trust. Their trust self-evaluation is based on monitoring and a challenge-response system.

Theodorakopoulos and Baras [3] [4], [5] analyze the issue of evaluating the trust level as a generalization of the shortest-path algorithm in a directed graph, where the edges correspond to the opinion that a node has about other node. They consider that nodes use just local information to establish their opinions. The opinion of each node includes the trust level and a value that represents the precision of the trust level. The main goal is to enable nodes to indirectly construct trust relationships using exclusively local information.

Sun *et al.* [6] have developed one framework capable of measuring the trust level and propagating it through the network. The goal is to secure routing and to assist intrusion detection systems. The framework also includes a defense mechanism against malicious nodes. They use a probabilistic model based on the uncertainty of a neighbor to execute one specific action and considers only local information.

We focus on providing nodes with a trust level for each direct neighbor, that is, neighbor within the radio range. The goal is to make nodes capable of gathering information to reason, learn, and make their own decisions. Different from most related works, our work improves scalability by restricting nodes to keep and exchange trust information solely with direct neighbors. We also introduce the concept of relationship maturity.

We present a trust model based on the human concept of trust. The model builds a trust relationship among the nodes of an ad hoc network. We have showed the correctness of our model in a single hop network [7]. In this paper, we analyze the behavior of the proposed model in a mobile multihop network. Results show that the maturity relationship improves the system performance allowing a higher mobility.

The paper is organized as follows. We present the main aspects of our trust model in Section II. Section III shows our simulation results. In Section IV we present our conclusions.

II. Trust model

The goal is to provide nodes with a mechanism to evaluate the trust level of its direct neighbors. Our model can be divided in two distinct layers as shown is Figure 1. The Learning layer is responsible for gathering and converting information into knowledge. The Trust layer defines how to assess the trust level of each neighbor using the knowledge information provided by the Learning layer and the information exchanged with direct neighbors. Both layers can interact with all layers of the TCP/IP model. In this paper, we focus on the Trust layer. In order to know how trustworthy a given neighbor is, each node assigns a so-called trust level for each direct neighbor. We propose a continuous representation for the trust level, ranging from 0 to 1 where 0 means the least reliable node and 1 means the most reliable node. Similar to the concept of human trust, the computation of the trust level of a given neighbor is based on previous experiences and also on the opinion of other neighbors about this specific neighbor. By previous experiences, we mean that a node keeps track of the good and bad actions taken by other neighbors. As a result, previous experiences allow nodes to have personal "opinion" about each of their neighbors. The Learning layer is the responsible for monitoring and judging other's neighbor actions. Neighbor nodes can further share their own opinions in order to improve the trust level evaluation. The transmission of a personal opinion about a specific node i is defined as a recommendation. Neighbor nodes take into account this recommendation while calculating the local trust level for node i. For that purpose, we introduce the concept of relationship maturity, which is based on the age of the relationship between two nodes. This concept allows nodes to give more importance to recommendations sent by long-term neighbors than the one sent by new neighbors. Nodes willing to consider the recommendation of other nodes use the proposed Recommendation Exchange Protocol (REP) to keep the trust level of each neighbor up to date [7]. We assume the existence of an authentication mechanism.

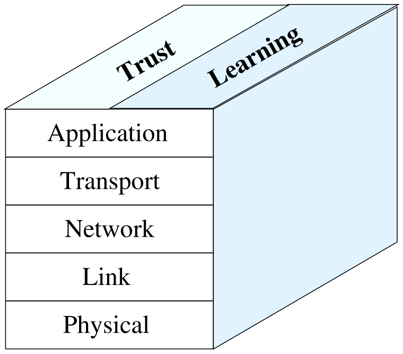


Fig. 1. Trust model.

A. Trust level evaluation

When a node first meets a new neighbor, it must assign an initial level of trust to this neighbor. This first value depends on the network condition, level of mobility, time, and place. Afterwards, the trust level evaluation process begins with a trust recommendation request and the monitoring of the new neighbor.

We define the trust level evaluation from node a about node b as a sum of its own trust and the contribution of other nodes, in the same way as defined by Virendra *et al.* [3]. The fundamental equation is

$$T_a(b) = (1-\alpha)Q_a(b) + \alpha C_a(b), \qquad (1)$$

where α permits choosing the most relevant factor. The variable $Q_a(b)$ represents the capability of a node to evaluate the trust level of their neighbors based on its own information. In order to obtain $Q_a(b)$, we propose the following equation

$$Q_a(b) = \beta E_T + (1-\beta)T_a(b), \qquad (2)$$

where E_T represents the value obtained by the judgment of a neighbor actions, and the variable β allows choosing which factor is the more relevant at a given moment.

B. Contribution computation

The trust level calculation also considers the recommendation of direct neighbors. The set of recommendations is called contribution ($C_a(b)$ in Equation 1). Recommendation can be obtained by sending a Trust Request (TREQ) or by receiving a Trust Advertisement (TA) message from other neighbors. TA messages are unsolicited recommendations. A node only sends a TA message when the recommendation about a particular neighbor varies more than a certain threshold value.

The contribution ($C_a(b)$) is defined as the sum of the recommendations from all nodes $i \in K_a$ about node b weighted by the trust level of node a about node i, as follows

$$C_a(b) = \frac{\sum_{i\in K_a} T_a(i)M_i(b)X_i(b)}{\sum_{j\in K_a} T_a(j)\sum_{j\in K_a} M_j(b)}. \qquad (3)$$

The group K_a defines the nodes from which recommendations will be considered. It is a subset of the neighbors of node a comprising all nodes that satisfy certain conditions. The contribution considers not only the trust level of others but also the accuracy and the relationship maturity. The accuracy of a trust level is defined by the standard deviation, similar to Theodorakopoulos and Baras [4]. The value in the trust level table of node a regarding node b is associated to a standard deviation $\sigma_a(b)$, which refers to the variations of the trust level that node a has observed about node b. We use X as a random variable with a normal distribution to represent the uncertainty of the recommendation. It can be expressed as

$$X_i(b) = N(T_i(b), \sigma_i(b)). \qquad (4)$$

The recommendation of node i about node b is weighted by $M_i(b)$, Let $M_i(b)$ be defined as the maturity of the relationship between nodes i and b, measured at node i. The relationship maturity is a measure of the time that two nodes have known each other. We use the relationship maturity to give more relevance to the nodes that know the evaluated neighbor for a long time. Accordingly, we assume that the trust level of a more mature neighbor has already converged to a common value within the network and therefore its opinion should be more relevant than the opinion of a new neighbor. We see in Section III that such assumption is valid. It is important to notice that maturity is only considered between the recommender (i) and the node that is being evaluated (b), namely, node a will never judge the opinions from neighbors that it knows longer more relevant.

Malicious nodes might try to fake trust levels for several reasons. One can try to slander a trustworthy node, to make other nodes believe that a specific malicious node can be trusted, or just to confuse other nodes. In order to minimize this effect, each node must define a maximum relationship maturity value M_{max}, which represents an upper bound for the relationship maturity. This value is based on the average time for which a node knows its neighbors.

III. Results

This section presents the results and the main characteristics of the simulator we have implemented to evaluate the proposed scheme. In ad hoc networks, nodes might perform several actions, like sending packets, forwarding packets, responding to routing messages, among others. The set of performed actions define the node behavior. Therefore, the learning layer monitors the neighbor actions trying to evaluate their behavior. In our home-made simulator, each node performs good actions and/or bad actions. Nodes perform actions according to an exponential distributed variable. The kind of action that will be performed depends solely on the nature of the node. A node with a nature equals to 0.8 means that it performs eight good actions out of ten.

The nature of a node ranges from 0 to 1. Most trustworthy nodes have nature equals to 1 while nodes untrustworthy have nature equals to 0. The nature is used as a reference of the ideal global trust level that a node should receive by its neighbors. We use it here as a metric to evaluate how close the measured global trust level of a node actually gets from its nature.

Another important characteristic introduced in our simulator is the perception of a node. The perception indicates the probability of noticing a certain action. Therefore, a node with 0.4 of perception is able of noticing 40% of all the actions performed by its neighbors. This parameter simulates an interaction between the learning layer and the trust layer, since the perception and the judgment of an action is the responsibility of the Learning layer. It is worth to mention that noticing and judging an action does not imply using promiscuous mode. We believe that a node should be able to decide whether it will use promiscuous mode or not based on its own constrains and needs. Thus, nodes might decide not to use promiscuous mode at the expense of having a lower perception.

The term that considers the experiences of the own node in Equation 2 is calculated using the last i perceived actions. It implies the existence of a minimum number of actions i that a node must notice from each neighbor to be able of having an opinion about them, based on its own experience. This means that during the initial phase of first contact, nodes use just the recommendations of its neighbors to evaluate the trust level of the new one.

Our main goal in this paper is to evaluate the trust system performance in mobile multihop networks. We are also interested in analyzing the impact of the relationship maturity and the influence of the variation of parameters α and perception. All results are presented with a confidence interval of 95% from a set of 100 replications. All figures present the trust level error (TLE) in time, which stands for the difference between the evaluated trust level and the correct value during the simulation time. The correct value is given by the nature of the node. In an ideal trust system the TLE would reach zero, which means that the node was able to perfectly evaluate the character of its neighbors.

The simulation scenario consists of 21 nodes with 250 m transmission range, which are placed in a 1000 m × 400 m area, as shows Figure 2. The distance between nodes is 150 m. We defined the first trust assignment equals to 0.9 for every node in the simulation. We also chose $\alpha = \beta = perception = 0.5$. These are the standard values for the simulations. For each specific configuration, the parameters that differ from its standard values are outlined. At last, in each configuration, all nodes have the same nature equals to 0.2. In the first configuration, node 8 moves away to a specific place and then, after a pause, comes back to its origin, and it goes back and forth during all simulation. Figure 3 presents the average TLE for all neighbors of node 8. The lower curve shows the result when node 8 goes to the same place as node 10 (m_1 in Figure 2) and the other one when it moves to node 12 (m_2). The main difference is the number of new neighbors. In the shorter movement (m_1), node 8 keeps 3 old neighbors while in m_2 all neighbors are new ones. We set the speed equals to 1 m/s and 2 m/s respectively. Thus, node 8 takes the same amount of time to move to both destinations. We observe in Figure 3(a) that the TLE begins in a certain level but tends to zero, but never reaches it. As we have noticed for singlehop networks in [7], there is a transient period and a stationary period. In the transient period nodes are trying to approximate to the expected value, while in the stationary period, the trust level is almost stable, very close to the correct value. When the simulation starts and nobody knows each other, we notice a certain transient, because the first trust assignment is 0.9 and the nature of all nodes is equal to 0.2. These initial values were chosen as worst case parameters. When node 8 moves away and meets new neighbors, we observe a peak in the TLE. This peak is lower than the first one, because node 8 receives, since its arrival at the new destination, "correct" recommendations from its new neighbors since they already know their old neighbors. The difference between the two curves is the destination place. The lower one represents a situation in which node 8 moves to a place where it already knows 3 neighbors, while the other one, it moves to a place where it knows nobody. Figure 3(b) shows the exactly scenario but node 8 moves three times faster. It is clear that in these conditions, node 8 does not stay long enough to evaluate the trust level of its neighbors.

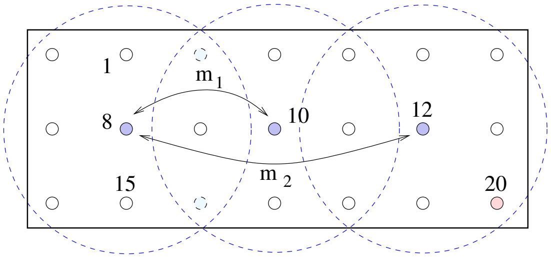
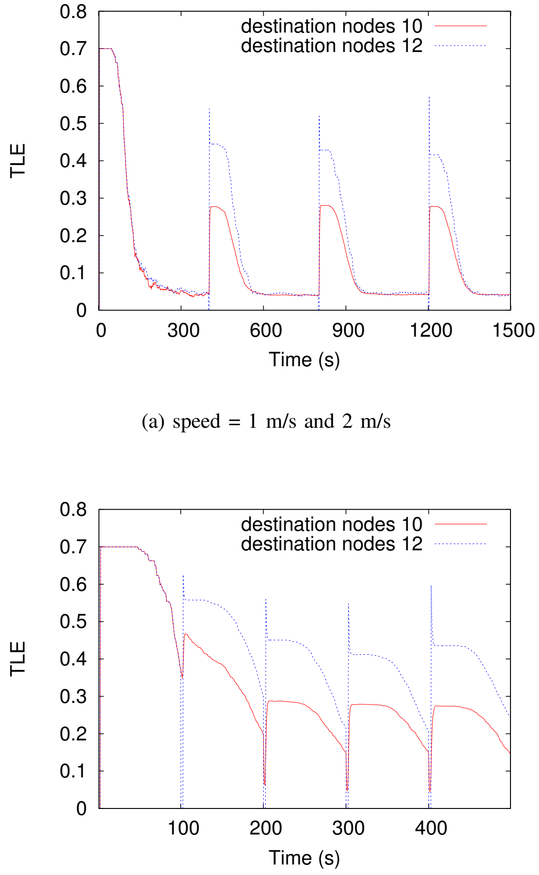


Fig. 2. Scenario.

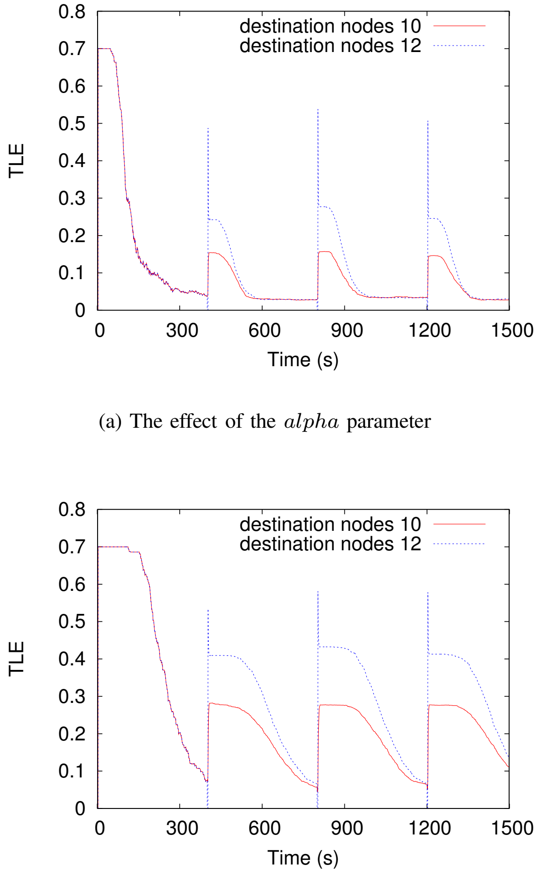
Figure 4 presents the results from the same movement pattern previously described (speed = 1 m/s and 2 m/s), and we vary the $alpha$ parameter and the perception of node 8. We can notice that increasing $alpha$ (Figure 4(a)) implies a decrease in the TLE during the transient period if compared with Figure 3(a). Figure 4(b) shows that if node 8, the one that moves, has a lower perception, it takes longer to reach the nature of its neighbors, which restrains the mobility. Afterwards, we analyze the impact of the relationship maturity in the evaluation of the trust level. For this purpose, we use a new configuration in the same scenario of Figure 2. In the new configuration, nodes 1,8,15 are going to move to the same place as node 12. Instead of



(a) speed = 1 m/s and 2 m/s

(b) speed = 3 m/s and 6 m/s without pause

Fig. 3. TLE in the presence of mobility with different velocities.

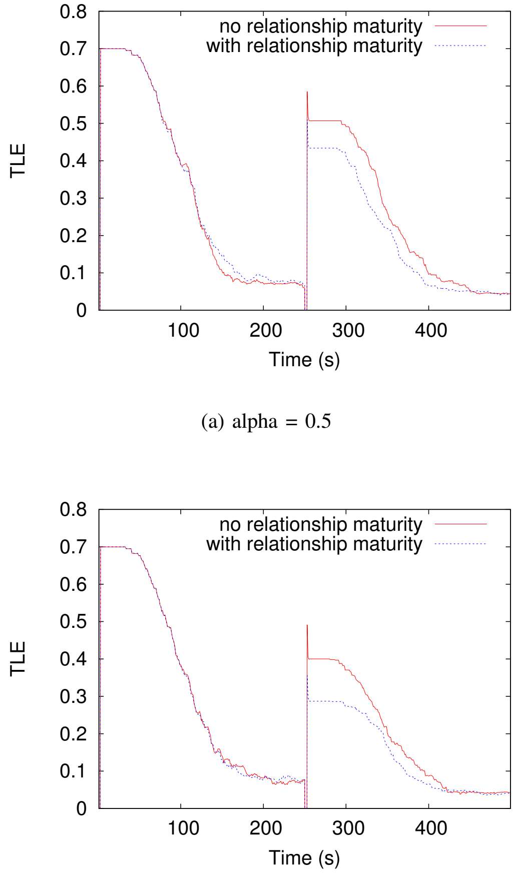


(a) The effect of the $alpha$ parameter

(b) The effect of perception

Fig. 4. TLE in the presence of mobility.

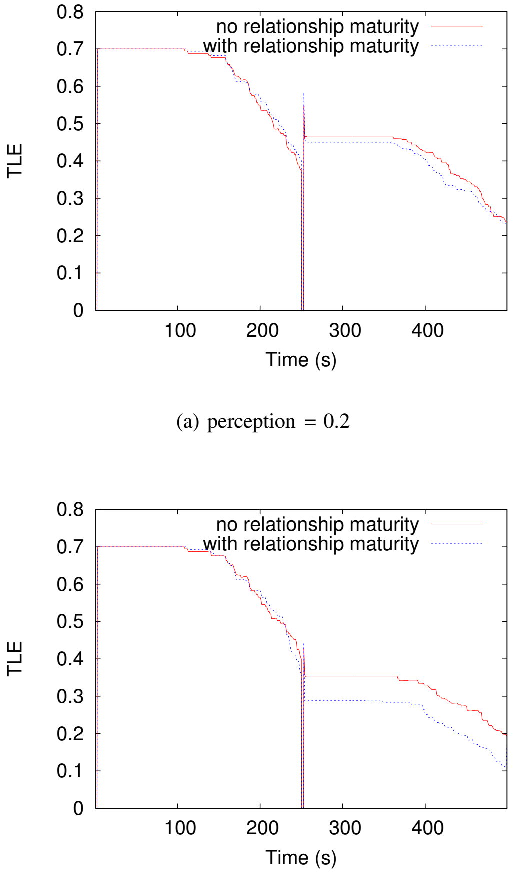
monitoring the trust level of all neighbors of node 8, we consider the trust level evaluation of node 8 about node 7 and node 20. Therefore, when node 8 arrives at the destination, nodes 1 and 15 have just arrived there. It means that node 20 has 3 new neighbors and 3 old ones. The old ones have a better idea about the nature of node 20 than the new ones. Without the relationship maturity, when node 8 receives the recommendations of its neighbors, it will treat them all the same manner. Using the relationship maturity allows node 8 to give more importance to the recommendations of the oldest neighbors of node 20. The result can be seen in Figure 5. It can be noticed by Figure 5(a) that using the relationship maturity the transient is shorter. Figure 5(b) shows that with a greater $alpha$ the impact of the relationship maturity in the transient is more significant. It improves the efficiency of the system due to the fact that node 8 prioritizes the recommendations of its neighbors. Figure 6(a) displays the impact of the relationship maturity when node 8 has a lower perception (0.2). We can observe that it presents a lower peak when node 8 arrives at the destination, but the difference is not significant as in the other figures. In this case, node 8 has a longer transient caused by the lower perception. It happens because trust updates are triggered only by actions, thus a difficulty of perception implies a longer transient. In Figure 6(b) we decreased the perception of node 8 and increased the value of $alpha$. It indicates that this is a good combination for a mobile network. Moreover, the effect of the relationship maturity is more evident. In this case when nodes have some difficulty to notice the actions of its neighbors, expressed by the low perception, the recommendations have greater importance. Therefore, valuing the recommendations from nodes that have a longer relationship with the node being evaluated is more effective. Although node 8 is not able to reach the stationary period, it achieves a lower TLE than without using the relationship maturity. The relationship maturity will play a more important role when nodes start



(a) alpha = 0.5

(b) alpha = 0.8

Fig. 5. The impact of the relationship maturity.



(a) perception = 0.2

(b) perception = 0.2 and alpha = 0.8

Fig. 6. The impact of the relationship maturity.

to change their nature.

IV. Conclusion

This paper analyzes a trust model for mobile ad hoc networks. We aim at building a trust relationship among nodes, confining the interactions to direct neighbors to better scale on mobile networks. Our concern is different from other works that focus strictly on security issues. We provide a mechanism for nodes to evaluate the trust level of their neighbors. We analyze through simulations the performance of the proposed model in a mobile multihop network. The results show that the maturity relationship improves the system performance allowing a higher mobility. We also show the impact of the main parameters on the trust evaluation in the proposed model. The maturity parameter can decrease the trust level error up to 50%.

References

[1] Z. Liu, A. W. Joy, and R. A. Thompson, “A dynamic trust model for mobile ad hoc networks,” in *IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS’04)*, (Suzhou, Chine), May 2004.

[2] A. A. Pirzada and C. McDonald, “Trust establishment in pure ad-hoc networks,” *Wireless Personal Communications: An International Journal*, vol. 37, no. 1-2, pp. 139–168, Apr. 2006.

[3] M. Virendra, M. Jadliwala, M. Chandrasekaran, and S. Upadhyaya, “Quantifying trust in mobile ad-hoc networks,” in *Proceedings of IEEE International Conference on Integration of Knowledge Intensive Multi-Agent Systems (KIMAS’05)*, (Waltham, USA), Apr. 2005.

[4] G. Theodorakopoulos and J. S. Baras, “Trust evaluation in ad-hoc networks,” in *Proceedings of the ACM Workshop on Wireless Security (WiSE’04)*, (Philadelphia, USA), Oct. 2004.

[5] G. Theodorakopoulos and J. S. Baras, “On trust models and trust evaluation metrics for ad hoc networks,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 318–328, Feb. 2006.

[6] Y. Sun, Z. Han, W. Yu, and K. J. R. Liu, “A trust evaluation framework in distributed networks: Vulnerability analysis and defense against attacks,” in *IEEE INFOCOM’06*, (Barcelona, Spain), Apr. 2006.

[7] P. B. Velloso, R. P. Laufer, O. C. M. B. Duarte, and G. Pujolle, “HIT: A human-inspired trust model,” in *8th IFIP IEEE International Conference on Mobile and Wireless Communication Networks - MWCN’2006*, (Santiago, Chile), Aug. 2006.